Meeting 2: Objects

Today

- Final Project
- Objects: Last Homework!

- Easier than HW5 ["4th late day"]

- Reference Compiler HW5 up.

Homework Tips

- Get one example through whole compiler (one-at-a-time)

  class C:

  |

- Looking at Quizzes

  Declassify

  "visit" case

- Don't Redoing work
Python Classes and Objects

- If you know Java or C++, forget it!

New Syntactic Elements ($P_3$)

expression ::= ... expression "(" expression "..." ")"

expression ::= expression "." identifier

attribute access of object or class

expression_list ::= ...

statement ::= ...

| "class" identifier
| ["(" expression_list ")"]

"." suite

suite ::= statement +

- define a class

" looks like def "
class C:

C = class lambda:

\[
\text{target} ::= \text{expression} \quad \text{identifier}
\]

\[
\text{simple statement} ::= \ldots
\]

\[
\text{target} ::= \ldots \text{ expression}
\]

C.x = 5

statement ::= \ldots

1 "while" expression ":" suite
1 "if" expression ":" suite
"else" ":" suite
Examples

class C:
    x = 42
print C.x

class C:
    print 4 * 10 + 2
    temp = create_class
    print 4 * 10 + 2
    C = temp

class C:
    f = lambda o, dx: o.x + dx
c . f

"unbound method"

"closure + restriction that the first arg must be an instance of this class"

class C:
    x = 5
D = C
D.x = 7
print C.x

o = C.o
(C.f)(o, 3)
"closure"
"unbound method"
class C:
    x = 42

class D:
    x = 0

print (C if input() else D).x

class A:
    x = y

class B:
    x = 0
    y = 2

class C(A, B):
    z = 3

print C.x * 10 + C.y

---

resolving attributes
depth-first,
left-to-right.
Objects

\[0 = C()\]

class C:
    def __init__(x, y):
        print(y)

\[o_1 = C(42)\]

\[o_2 = C(42)\]